WHAT IS CLAIMED IS:

1. A method for maximizing throughput while avoiding overload of one or more servers, comprising the steps of:

intercepting, via an interface unit, a client request for information from the server;

determining, by said interface unit, the current server performance, wherein said server performance is based on one or more of: the number of active connections opened to the server, the response time of the server and the rate at which said response time is changing;

forwarding said client request to the server if said current server performance is below or within a range determined for optimal performance, whereby avoiding overload of the server; and

where server performance is outside such optimal range, buffering the client request until said current server performance is within the optimal range for server performance.

2. The method of claim 1, wherein said buffering utilizes a first-in-first-out method.

3. The method of claim 1, wherein said buffering comprises the steps of:

determining a preferred client value for said client request; and determining the position of said client request in a queue based on a preferred client value.

4. The method of claim 3, wherein said preferred client value may be partly determined by one or more of the network address (including either or both of the internet address and the port address) of said client request, by a

header related to said client request, by previous requests from the client of said client request, and by a cookie related to said client request.

5. The method of claim 1, further comprising the step of multiplexing connections to the server, whereby said multiplexed connections may be reused for different client requests.

6. The method of claim 1, further comprising the step of closing connections to the server as a way of reducing server load and improving server performance.

7. The method of claim 1, wherein the step of determining the current server performance may further be determined by the number of pending requests sent to the server and server error/overload messages from the server.

8. A system for maximizing throughput while avoiding overload of a server, comprising an interface unit for intercepting a client request for information from the server,

wherein said interface unit determines the current server performance based on the number of connections opened to the server, the response time of the server and the rate at which said response time is changing, wherein said interface unit forwards said client request to the server if said current server performance is below or within a range determined for optimal performance, whereby avoiding overload of the server, and

where server performance is beyond such optimal range, wherein said interface unit buffers the client request until said current server performance is within the optimal range for server performance.

9. The system of claim 8, wherein said interface unit buffers said client request by utilizing a first-in-first-out method.

10. The system of claim 8, wherein said interface unit buffers said client request by determining a preferred client value for said client request and determining the position of said client request in a queue based on a preferred client value.

11. The system of claim 10, wherein said preferred client value may be partly determined by one or more of the network address (including either or both of the internet address and the port address) of said client request, by a header related to said client request, by previous requests from the client of said client request, and by a cookie related to said client request.

12. The system of claim 8, wherein said interface unit multiplexes connections to the server, whereby said multiplexed connections may be reused for client requests from different clients.

13. The system of claim 8, wherein said current server performance may further be determined by the number of pending requests sent to the server and server error/overload messages from the server.